模块5 人工智能边缘设备计算框架


理论讲解

随着深度学习技术的逐步成熟和日益普及，模块化、标准化的流程工具成为开发者的普遍诉求，深度学习框架应运而生。深度学习框架提供多种基础功能的算法库，帮助开发者将有限精力专注于更高层级的创新突破，实现在巨人肩膀上的创新。

【模块描述】

本模块主要讲解人工智能边缘计算技术栈中，智能边缘设备计算框架的知识与应用。作为 AI 基础技术，深度学习框架能够集训练和推理框架、开发套件、基础模型库、工具组件于一体，提供由高级语言封装的多样化接口，实现快速便捷的关键模型构建、训练和调用，利用工具化、平台化的方式帮助广大开发者和企业进一步降低深度学习技术应用门槛，加速行业智能化转型。

本次项目主要学习目前行业内常用的深度学习开发框架，包括 Paddle Inference、TensorFlow Lite、MNN、NCNN、OpenVINO 等。在实操部分，将通过安装 Paddle Inference，并对安装情况进行验证，掌握不同系统下深度学习开发框架的环境部署方法。

【学习目标】

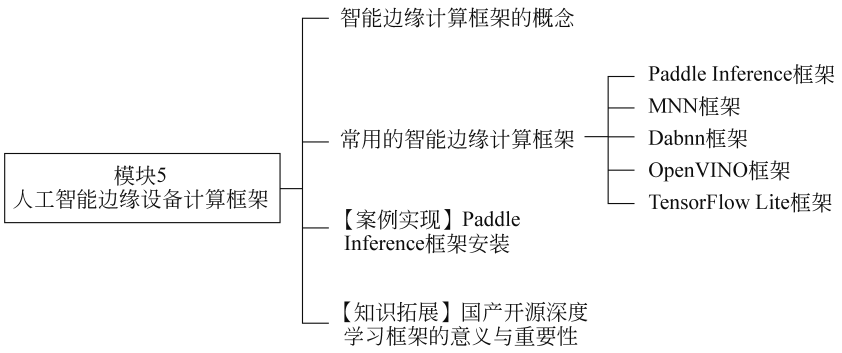
知识目标	能力目标	素质目标
(1) 了解深度学习计算框架的概念。 (2) 熟悉常用的深度学习计算框架。	(1) 能够搭建深度学习开发框架所需的环境。 (2) 能够部署深度学习开发框架。	国产开源深度学习框架的意义与重要性。

【课程思政】

结合操作系统的发展历程及国内外发展现状，如开源鸿蒙操作系统等，客观分析我国操作系统内核短板尚未突破的现实困局，讨论聚焦桌面操作系统业已取得长足进步并开花结果的发展态势，以及中美科技角逐的态势，激发学生勇于担当、积极投身国家重大需求的爱国热情。结合操作系统发展中的典型实例，适时引入操作系统领域图灵奖获得

者及贡献,激发学生勇于创新、积极探索的科学精神;结合典型操作系统譬如 Windows、Linux 发展过程中的成功经验,培养学生辩证思维、知行合一、精益求精、与时俱进、团队协作的意识与能力。

【知识框架】



【知识准备】

5.1 智能边缘计算框架的概念

智能边缘计算框架是部署在边缘设备等小型移动设备上的深度学习框架,能够使得一些大模型的推理任务可以在设备本地执行。智能边缘计算框架主要包括模型优化器和推理引擎两个部分。

模型优化器(Model Optimizer)是一个跨平台的命令行工具,能够将深度学习训练框架如 PaddlePaddle、Caffe、TensorFlow、PyTorch 等框架训练后的模型转换为部署所需要的模型,其主要分为三个部分的工作。

(1) 转换:转换为统一的 IR 格式档案。

(2) 优化:根据不同设备不同优化方法节省计算时间和存储器空间。

(3) 转换权重与偏置:根据需求转换权值不同的模型格式。

而推理引擎则是将逻辑规则应用于知识库以推断出新信息的系统组件,主要工作是针对特定目标设备进行优化来完成推理功能。

5.2 常用的智能边缘计算框架

常用的智能边缘计算框架有 Paddle Inference、TensorFlow Lite、MNN、Dabnn、OpenVINO 等。

5.2.1　Paddle Inference 框架

Paddle Inference 是百度飞桨 PaddlePaddle 深度学习框架的原生推理库，为人工智能边缘服务提供高性能的推理能力。由于其能力直接基于 Paddle 的训练算子，因此 Paddle Inference 可以通用支持 PaddlePaddle 训练出的所有模型。Paddle Inference 功能特性丰富，性能优异，针对不同平台不同的应用场景进行了深度的适配优化，可做到高吞吐、低时延，保证了 PaddlePaddle 模型在服务器端即训即用，快速部署。其高性能主要通过以下几个方面来实现。

(1) 内存和显存复用提升服务吞吐量。Paddle Inference 在推理初始化阶段，对模型中的 OP 输出 Tensor(张量)进行依赖分析，将两两互不依赖的 Tensor 在内存和显存空间上进行复用，进而增大计算并行量，提升服务吞吐量。

(2) 细粒度 OP 横向纵向融合减少计算量。Paddle Inference 在推理初始化阶段，按照已有的融合模式将模型中的多个 OP 融合成一个 OP，减少了模型计算量的同时，也减少了 Kernel Launc 的次数，从而能提升推理性能。

(3) 子图集成 TensorRT 加快 GPU 推理速度。Paddle Inference 采用子图的形式集成 TensorRT，针对 GPU 推理场景，TensorRT 可对一些子图进行优化，包括 OP 的横向和纵向融合，过滤冗余的 OP，并为 OP 自动选择最优的 kernel，加快推理速度。

(4) 子图集成 Paddle Lite 轻量化推理引擎。Paddle Lite 是飞桨深度学习框架的一款轻量级、低框架开销的推理引擎，除了在移动端应用外，还可以使用服务器进行 Paddle Lite 推理。Paddle Inference 采用子图的形式集成 Paddle Lite，以方便用户在服务器推理原有方式上稍加改动，即可开启 Paddle Lite 的推理能力，得到更快的推理速度。并且，使用 Paddle Lite 可支持在百度昆仑等高性能人工智能计算芯片上执行推理计算。

(5) 支持加载 PaddleSlim 量化压缩后的模型。PaddleSlim 是飞桨深度学习模型压缩工具，Paddle Inference 可联动 PaddleSlim，支持加载量化、裁剪和蒸馏后的模型并部署，由此减小模型存储空间、减少计算占用内存、加快模型推理速度。其中，在模型量化方面，Paddle Inference 在 x86 CPU 上做了深度优化，常见分类模型的单线程性能可提升近 3 倍，ERNIE 模型的单线程性能可提升 2.68 倍。

在通用性层面，Paddle Inference 不仅与主流软硬件环境兼容适配，支持服务器端 x86 CPU、NVIDIA GPU 芯片，兼容 Linux/macOS/Windows 系统，支持所有飞桨训练产出的模型，完全做到即训即用，而且拥有多语言环境的丰富接口并供灵活调用，支持 C++、Python、C、Golang，接口简单灵活，20 行代码即可完成部署。对于其他语言，提供了应用程序二进制接口(ABI)和稳定的 C 语言应用程序编程接口，用户可以很方便地扩展。

5.2.2　MNN 框架

MNN(Mobile Neural Network)是阿里巴巴淘系技术开源的一个轻量级的深度神经网络推理引擎。MNN 支持深度模型推理与训练，尤其在边缘加载深度神经网络模型进行推理预测。其整体有以下 4 大特性。

(1) 轻量性。MNN 针对边缘设备特点深度定制和裁剪，无任何依赖，可以方便地部署

到移动设备和各种嵌入式设备中。

(2) 通用性。MNN 支持 Paddle、TensorFlow、Caffe、ONNX 等主流模型文件格式,支持卷积神经网络(CNN)、循环神经网络(RNN)、对抗生成网络(GAN)等常用网络,支持 86 个 TensorFlow 算子,34 个 Caffe 算子。同时,MNN 还支持异构设备混合计算,目前支持 CPU 和 GPU 混合,可以动态导入 GPU 算子插件,替代 CPU 算子的实现。

(3) 高性能。MNN 不依赖任何第三方计算库,依靠大量手写汇编实现核心运算,充分发挥 ARM CPU 的算力。MNN 能够高效稳定地实现人工智能程序中卷积、循环等深度学习算法,对于任意形状的卷积均能高效运行。

(4) 易用性。MNN 有高效的图像处理模块,覆盖常见的形变、转换等需求。一般情况下,无须额外引入 libyuv 或 OpenCV 库处理图像。

5.2.3 Dabnn 框架

二值神经网络 (BNN) 是一种特殊的神经网络,它将网络的权重和中间特征压缩为 1b,可以看作普通浮点型网络量化到极致的结果。和其他比特数稍高的量化网络(例如三值网络、2-bit 网络、4-bit 网络)相比,二值网络最突出的优点在于,1-bit 乘加操作可以通过位运算高效实现,因此可以无缝运行在主流硬件平台(x86、ARM)上。

二值神经网络在边缘设备上具有巨大潜力,因为它们通过高效的逐位运算取代了浮点运算。而 Dabnn 框架是京东推出的一个高度优化的移动平台二进制神经网络推理框架。使用 ARM 汇编实现了二进制卷积。

5.2.4 OpenVINO 框架

OpenVINO 是英特尔针对板载英特尔芯片的硬件平台开发的一套深度学习工具库,包含推断库、模型优化等一系列与深度学习模型部署相关的功能。OpenVINO 工具包是用于快速开发应用程序和解决方案的综合工具包,可解决各种任务,包括模拟人类视觉、自动语音识别、自然语言处理、推荐系统等。该工具包基于最新一代的人工神经网络,包括卷积神经网络、循环神经网络和基于注意力的神经网络,可在 Intel 硬件上扩展计算机视觉和非视觉工作负载,从而最大限度地提高性能。

OpenVINO 是一个比较成熟且仍在快速发展的推理库,可以用来快速部署开发,尤其是板载英特尔芯片的硬件平台上性能超过了大部分的开源库。OpenVINO 对各类图形图像处理算法进行了针对性的优化,从而扩展了 Intel 的各类算力硬件以及相关加快器的应用空间,实现了 AI 范畴的异构较量,使传统平台的视觉推理能力获得了很大水平的提高。

5.2.5 TensorFlow Lite 框架

TensorFlow Lite 是一款专门针对移动设备的深度学习框架。它使设备机器学习具有低延迟和更小的体积,可以使用训练好的模型在人工智能边缘设备上完成推理任务。

TensorFlow Lite 支持一系列量子和浮点的核心运算符,并针对移动平台进行了优化。它结合 pre-fused 激活和其他技术来进一步提高性能和量化精度。此外,TensorFlow Lite 还支持在模型中使用自定义操作。

TensorFlow Lite 拥有一个新的移动设备优化的解释器，保证人工智能程序的精简和快速应用。解释器是一种能够把高级编程语言一行行直接转译运行的计算机程序，而在本门课程学习中所使用的 Python 语言也是需要解释器来执行的，解释器与设备的匹配度，直接表现为代码运行速度。TensorFlow Lite 所配备的解释器使用静态图形排序和自定义内存分配器来确保最小的负载，保证程序和设备运行效率。


实操讲解

【案例实现】 Paddle Inference 框架安装

基于模块描述与知识准备的内容，在基本了解深度学习框架的作用及其常用的深度学习框架后，接下来在 Linux 环境下编译 Paddle Inference 源码，生成目标硬件为 Linux 的预测库，掌握 Paddle Inference 在 Linux 中的部署方法。

本次案例实训的思路如下。

（1）查看人工智能边缘设备硬件环境。通过相关命令查看人工智能边缘设备的硬件环境，包括 Python 版本以及硬件架构。

（2）验证人工智能边缘设备编译环境。验证人工智能边缘设备当前的编译环境是否符合框架安装要求。

（3）安装 Paddle Inference 框架。通过命令行的方式安装 Paddle Inference 框架。

（4）验证 Paddle Inference 安装情况。安装完 Paddle Inference 后执行相关脚本进行环境验证。

任务 1：查看人工智能边缘设备硬件环境

接通边缘设备电源，通过本地连接或者远程连接的方式进入边缘设备的桌面，在边缘设备的桌面中单击右键，选择 Open Terminal 选项打开终端命令行，如图 5-1 所示。

图 5-1　打开终端命令行

在打开的终端命令行中依次输入以下命令,即可查看当前环境下的 Python 版本以及硬件环境。

```
python --version
python -c "import platform;print(platform.architecture()[0]);print(platform.machine())"
```

在终端命令行中输入上述命令后,即可在终端命令行中查看输出的 Python 版本以及人工智能边缘设备的硬件架构等信息,如图 5-2 所示。

图 5-2 查看安装环境

任务 2:验证人工智能边缘设备编译环境

在确定当前硬件环境后,需要准备并验证当前的编译环境,编译 Paddle Inference 的环境要求有 python-pip(版本为 20.2.2 或更高)、python-numpy(1.18.1 或更高)。

在明确要求后,继续在终端命令行中依次输入以下命令,即可查看并更新 pip 以及 numpy 的版本。

```
Python3 -m pip --version
pip3 install --ungrage pip
pip install --upgrade numpy
```

在终端命令行中输入上述命令后,即可在终端命令行中查看输出的 pip 以及 numpy 的版本信息,如图 5-3 所示。

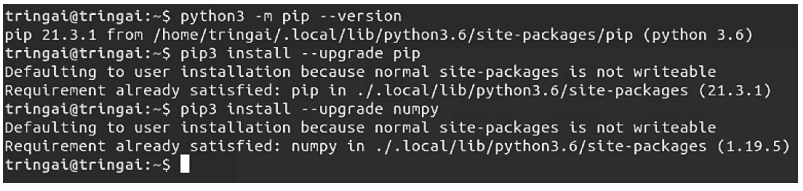


图 5-3 查看 pip 和 numpy 的版本信息

确定好当前环境满足框架安装的条件后,接下来便可以对 Paddle Inference 框架进行安装。

任务 3:安装 PaddlePaddle 框架

首先需要确定当前设备的 Jetpack 版本,在终端命令行中输入以下命令即可查看对应的 Jetpack 版本。

```
cat /etc/nv_tegra_release      #查看 Jetpack 版本
```

在终端命令行中输入上述命令后，即可在终端命令行中查看 Jetpack 的版本，结果如图 5-4 所示。

```
tringai@tringai:~$ cat /etc/nv_tegra_release
# R32 (release), REVISION: 5.2, GCID: 27767740, BOARD: t210ref, EABI: aarch64, DATE: Fri Jul  9 16:01:52 UTC 2021
tringai@tringai:~$
```

图 5-4　查看 Jetpack 版本

接着继续在终端命令行中输入以下命令，安装对应的 PaddlePaddle 框架。

```
pip3 install https://paddle-inference-lib.bj.bcebos.com/2.3.0/python/Jetson/
jetpack4.5_gcc7.5/nano/paddlepaddle_gpu-2.3.0-cp36-cp36m-linux_aarch64.whl
```

在终端命令行中输入上述安装命令后，即可在终端中看到安装完成的字样，如图 5-5 所示。

```
tringai@tringai:~$ pip3 install https://paddle-inference-lib.bj.bcebos.com/2.3.0/python/Jetson/jetpack4.5_gcc7.5/nano/paddlepaddle_gpu-2.3.0-cp36-cp36m-linux_aarch64.whl
Defaulting to user installation because normal site-packages is not writeable
Collecting paddlepaddle-gpu==2.3.0
  Downloading https://paddle-inference-lib.bj.bcebos.com/2.3.0/python/Jetson/jetpack4.5_gcc7.5/nano/paddlepaddle_gpu-2.3.0-cp36-cp36m-linux_aarch64.whl (142.2 MB)
     |████████████████████████████████| 142.2 MB 39 kB/s
Requirement already satisfied: numpy>=1.13 in ./.local/lib/python3.6/site-packages (from paddlepaddle-gpu==2.3.0) (1.19.5)
Requirement already satisfied: decorator in ./.local/lib/python3.6/site-packages (from paddlepaddle-gpu==2.3.0) (4.4.2)
Collecting opt-einsum==3.3.0
  Downloading opt_einsum-3.3.0-py3-none-any.whl (65 kB)
     |████████████████████████████████| 65 kB 468 kB/s
Requirement already satisfied: Pillow in ./.local/lib/python3.6/site-packages (from paddlepaddle-gpu==2.3.0) (8.4.0)
Requirement already satisfied: six in ./.local/lib/python3.6/site-packages (from paddlepaddle-gpu==2.3.0) (1.16.0)
Requirement already satisfied: protobuf>=3.1.0 in ./.local/lib/python3.6/site-packages (from paddlepaddle-gpu==2.3.0) (3.19.6)
Collecting paddle-bfloat==0.1.2
  Downloading paddle_bfloat-0.1.2-cp36-cp36m-manylinux_2_17_aarch64.manylinux2014_aarch64.whl (389 kB)
     |████████████████████████████████| 389 kB 1.4 MB/s
Requirement already satisfied: requests>=2.20.0 in ./.local/lib/python3.6/site-packages (from paddlepaddle-gpu==2.3.0) (2.27.1)
Requirement already satisfied: astor in ./.local/lib/python3.6/site-packages (from paddlepaddle-gpu==2.3.0) (0.8.1)
Requirement already satisfied: idna<4,>=2.5 in /usr/lib/python3/dist-packages (from requests>=2.20.0->paddlepaddle-gpu==2.3.0) (2.6)
Requirement already satisfied: certifi>=2017.4.17 in /usr/lib/python3/dist-packages (from requests>=2.20.0->paddlepaddle-gpu==2.3.0) (2018.1.18)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in ./.local/lib/python3.6/site-packages (from requests>=2.20.0->paddlepaddle-gpu==2.3.0) (1.26.13)
Requirement already satisfied: charset-normalizer~=2.0.0 in ./.local/lib/python3.6/site-packages (from requests>=2.20.0->paddlepaddle-gpu==2.3.0) (2.0.12)
Installing collected packages: paddle-bfloat, opt-einsum, paddlepaddle-gpu
  Attempting uninstall: paddlepaddle-gpu
    Found existing installation: paddlepaddle-gpu 2.0.2
    Uninstalling paddlepaddle-gpu-2.0.2:
      Successfully uninstalled paddlepaddle-gpu-2.0.2
Successfully installed opt-einsum-3.3.0 paddle-bfloat-0.1.2 paddlepaddle-gpu-2.3.0
tringai@tringai:~$
```

图 5-5　安装 PaddlePaddle 框架

在安装完成后，需要对安装情况进行验证，在终端命令行中依次输入以下命令，先对人工智能边缘设备进行设置，防止出现死机的情况。

```
sudo nvpmodel -m 0 && sudo jetson_clocks #打开性能模式
#增加 swap 空间，防止爆内存
sudo swapoff -a
sudo fallocate -l 15G /swapfile
sudo chmod 600 /swapfile
sudo mkswap /swapfile
```

```
sudo swapon /swapfile
sudo swapon -a
sudo swapon -show          #用来查看结果
ulimit -n 2048             #最大的文件打开数量
```

设置完成后,接着在终端命令行中输入“Python3”进入 Python 编辑器,进入编辑器后依次输入以下代码,即可查看 PaddlePaddle 框架的安装情况。

```
import paddle
paddle.fluid.install_check.run_check()
```

在 Python 编辑器中输入以上代码并运行后,即可看到如图 5-6 所示信息,表明 PaddlePaddle 框架已经安装完成。

```
tringai@tringai:~$ python3
Python 3.6.9 (default, Nov 25 2022, 14:10:45)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import paddle
>>> paddle.fluid.install_check.run_check()
Running Verify Fluid Program ...
W0117 11:06:49.918761 16005 gpu_context.cc:278] Please NOTE: device: 0, GPU Compute Capability: 5.3, Driver API Version: 10.2, Runtim
e API Version: 10.2
W0117 11:06:50.056411 16005 gpu_context.cc:306] device: 0, cuDNN Version: 8.0.
Your Paddle Fluid works well on SINGLE GPU or CPU.
I0117 11:08:03.861820 16005 parallel_executor.cc:486] Cross op memory reuse strategy is enabled, when build_strategy.memory_optimize
= True or garbage collection strategy is disabled, which is not recommended
Your Paddle Fluid works well on MUTIPLE GPU or CPU.
Your Paddle Fluid is installed successfully! Let's start deep Learning with Paddle Fluid now
>>>
```

图 5-6　验证 PaddlePaddle 安装情况

任务 4：验证 Paddle Inference 安装情况

接下来进行 Paddle Inference 安装情况的验证。首先在人工智能边缘设备桌面重新打开一个新的终端命令行,接着输入以下命令,即可切换到本次案例的文件夹目录。

```
cd Desktop/projects/char5/
```

接着输入以下命令对文件夹中的 Paddle-Inference-Demo-master.zip 压缩包进行解压。

```
unzip -oq Paddle-Inference-Demo-master.zip
```

解压完成后,即可运行文件中的测试脚本对安装环境进行验证,在运行测试脚本之前需要对脚本文件进行修改,在终端命令行中依次输入以下命令,即可对脚本文件进行编辑。

```
cd Paddle-Inference-Demo-master/python/cpu/resnet50
gedit run.sh
```

在终端命令行中输入上述命令并运行后,即可弹出一个文本编辑框对脚本文件进行编辑,如图 5-7 所示,此处只需要将最后一行的“python”修改为“python3”即可,修改完成后按 Ctrl+S 组合键或单击 Save 按钮即可保存修改的内容,随后单击关闭窗口按钮即可完成脚

本文件内容修改。

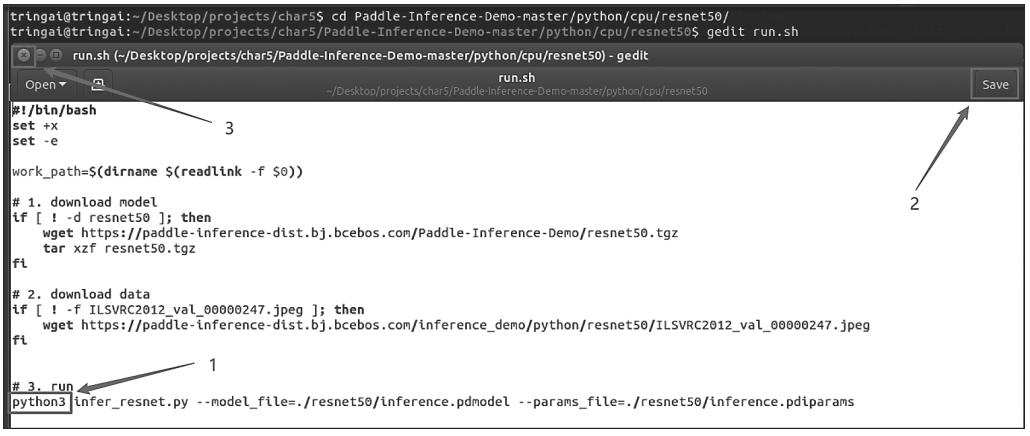


图 5-7　修改脚本文件

脚本文件修改完成后，接下来可以在终端命令行中依次输入以下命令，即可执行测试脚本文件，以验证 Paddle Inference 的安装情况。

```
chmod +x run.sh
./run.sh
```

在终端命令行中输入上述命令后，即可执行测试脚本文件，程序运行后将会下载 resnet50 模型和一张测试图片，接着调用模型对图片进行预测，如图 5-8 所示。

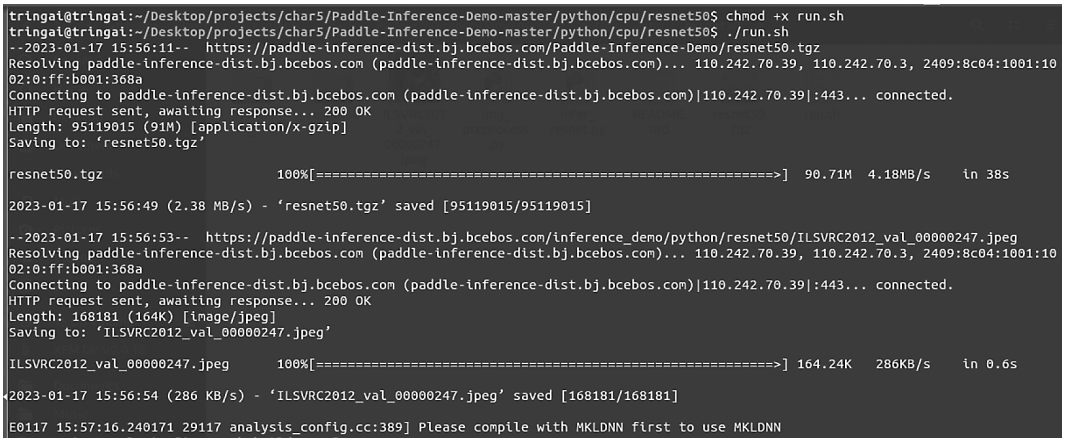


图 5-8　执行预测脚本文件

等待预测脚本运行完成后，可以看到预测当前图片所属的标签类别为 13，如图 5-9 所示。

```
---    fused 0 pairs of fc gru patterns
--- Running IR pass [mul_gru_fuse_pass]
--- Running IR pass [seq_concat_fc_fuse_pass]
--- Running IR pass [gpu_cpu_squeeze2_matmul_fuse_pass]
--- Running IR pass [gpu_cpu_reshape2_matmul_fuse_pass]
I0117 15:57:17.051901 29117 fuse_pass_base.cc:57] ---  detected 1 subgraphs
--- Running IR pass [gpu_cpu_flatten2_matmul_fuse_pass]
--- Running IR pass [matmul_v2_scale_fuse_pass]
--- Running IR pass [gpu_cpu_map_matmul_v2_to_mul_pass]
--- Running IR pass [gpu_cpu_map_matmul_v2_to_matmul_pass]
--- Running IR pass [matmul_scale_fuse_pass]
--- Running IR pass [gpu_cpu_map_matmul_to_mul_pass]
--- Running IR pass [fc_fuse_pass]
I0117 15:57:17.093112 29117 fuse_pass_base.cc:57] ---  detected 1 subgraphs
--- Running IR pass [repeated_fc_relu_fuse_pass]
--- Running IR pass [squared_mat_sub_fuse_pass]
--- Running IR pass [conv_bn_fuse_pass]
I0117 15:57:17.430953 29117 fuse_pass_base.cc:57] ---  detected 53 subgraphs
--- Running IR pass [conv_eltwiseadd_bn_fuse_pass]
--- Running IR pass [conv_transpose_bn_fuse_pass]
--- Running IR pass [conv_transpose_eltwiseadd_bn_fuse_pass]
--- Running IR pass [is_test_pass]
--- Running IR pass [runtime_context_cache_pass]
--- Running analysis [ir_params_sync_among_devices_pass]
--- Running analysis [adjust_cudnn_workspace_size_pass]
--- Running analysis [inference_op_replace_pass]
--- Running analysis [memory_optimize_pass]
I0117 15:57:17.556871 29117 memory_optimize_pass.cc:216] Cluster name : inputs  size: 602112
I0117 15:57:17.557394 29117 memory_optimize_pass.cc:216] Cluster name : batch_norm_13.tmp_2  size: 1605632
I0117 15:57:17.557742 29117 memory_optimize_pass.cc:216] Cluster name : elementwise_add_2  size: 3211264
I0117 15:57:17.558076 29117 memory_optimize_pass.cc:216] Cluster name : batch_norm_7.tmp_2  size: 3211264
I0117 15:57:17.558410 29117 memory_optimize_pass.cc:216] Cluster name : elementwise_add_1  size: 3211264
--- Running analysis [ir_graph_to_program_pass]
I0117 15:57:17.930994 29117 analysis_predictor.cc:1007] ======= optimize end =======
I0117 15:57:17.948957 29117 naive_executor.cc:102] ---  skip [feed], feed -> inputs
I0117 15:57:17.963081 29117 naive_executor.cc:102] ---  skip [save_infer_model/scale_0.tmp_1], fetch -> fetch
class index:  13
tringai@tringai:~/Desktop/projects/char5/Paddle-Inference-Demo-master/python/cpu/resnet50$
```

图 5-9　图片预测结果输出

【模块小结】

本模块首先介绍了人工智能边缘计算技术栈中边缘计算框架的概念和作用,接着从如今常用的深度学习计算框架着手,分别介绍了 Paddle Inference、MNN、Dabnn、OpenVINO、TensorFlow Lite 等框架的基本信息、目标设备以及主要特性。最后,以 Paddle Inference 框架的安装和验证为例,熟悉了深度学习推理框架环境的基本搭建流程以及框架的安装和验证过程。

【知识拓展】 国产开源深度学习框架的意义与重要性

2020 年对于中国科技行业来说是一个觉醒的元年。地缘政治产生的冲击波,警醒了许多国内的科技行业参与者。自主创新的呼声一浪高过一浪。在深度学习领域,Google 的 TensorFlow、Facebook 的 PyTorch 作为主流框架自然大名鼎鼎,中国企业近期也纷纷发布自己的开源框架,例如,旷视的 MegEngine、华为的 MindSpore、清华的 Jittor、一流的 Oneflow 等,加上最早开源且技术成熟、框架完备的百度飞桨 PaddlePaddle,可以说,这条赛道已经是风起云涌,形成了群雄逐鹿的局面。

现阶段,人工智能技术高速发展,推动着全球科技革命和产业变革,人类社会正在大步迈向智能时代。深度学习是新一代人工智能的关键技术,让很多此前无法实现的 AI 应用在现实生活中“跑起来”。例如,现在许多制造企业已经在深度学习的帮助下,打造了可以自动识别瑕疵零件的生产线,人工智能可以像人一样,发现零件上的“不合格”特征并指出来,